

Universidade de Caxias do Sul
 Área do Conhecimento de Ciências Exatas e Engenharias
 Disciplina de Fundamentos de Banco de Dados
 Professor Daniel Luis Notari
 Avaliação I - 03/10/2023

Aluno(a): Edmario Eberhardt Pereira 9,0 + 1,0 = 10,0

* **Questão 1. (1,0)** Leia a sentença abaixo, diga se é verdadeira ou falsa e justifique a sua resposta. Se for falsa, circule as partes que tornam a sentença falsa:

Um Sistema de Gerência de Banco de Dados (SGBD) é uma coleção de programas que permite aos usuários criarem e manter um banco de dados. O SDBD fornece algumas operações, tais como, especificação dos tipos de dados; armazenamento dos dados; e, permite manipulações de informações (consultas, atualização, geração de relatórios). Seu maior objetivo é proporcionar um ambiente conveniente e eficiente para recuperação e armazenamento das informações do SGBD.

A afirmação está correta. Ela define de forma correta as funções e objetivos do SGBD e do SDBD.

Questão 2. (1,0) Você foi contratado para projetar a estrutura de um banco de dados para uma nova rede social. Explique como você criaria as entidades e atributos necessários para armazenar informações sobre as conexões entre usuários, e garantir que a rede possa escalar conforme mais usuários aderem ao serviço. Justifique a sua escolha.

Eu criaria uma entidade de usuário que possui um auto-relacionamento para representar as requisições (um usuário segue o outro). Um post seria uma entidade que possui relação de posse com o usuário. Ligada ao post, a entidade curtidas seria ligada em relação ao post. Essa estrutura permite manter cada curtida de cada post e o respectivo usuário que curtiu, através do relacionamento usuário-curtida. Esse modelo ainda permite consultar todos os posts que um usuário curtiu, todos usuários que ele segue e todos que lhe seguem, dando alta escalabilidade e alto desempenho para as interações entre os usuários.

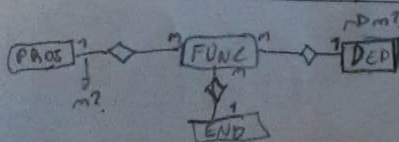
Questão 3. (1,0) Suponha que você tem uma tabela de funcionários com os seguintes atributos: ID do Funcionário, Nome, Departamento, Endereço, Salário e Projeto Atual. Aplique a normalização até a terceira forma normal (3FN) e descreva as tabelas resultantes, identificando as chaves primárias e os relacionamentos entre elas.

Funcionários (ID, nome, # cad - ender, salário, # cad - dep, # cad - proj)

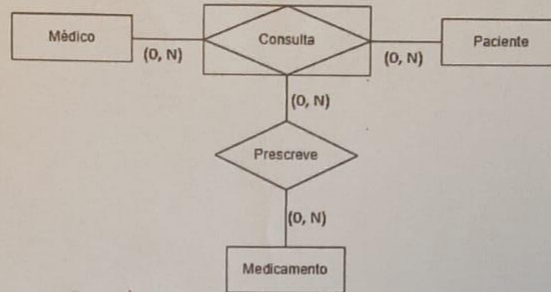
Endereço (ID, cidade, bairro, rua, num, apt, complemento)

Departamento (ID, nome)

Projeto (ID, nome, descrição, data - início, data - fim)



Questão 4. (3,0) Aplique as regras de tradução do Diagrama Entidade-Relacionamento para tabelas para o DER abaixo. A saída deve ser o modelo lógico indicando as chaves primárias sublinhadas e as chaves estrangeiras marcas com #. Defina os atributos e identificadores e mais para cada entidade.



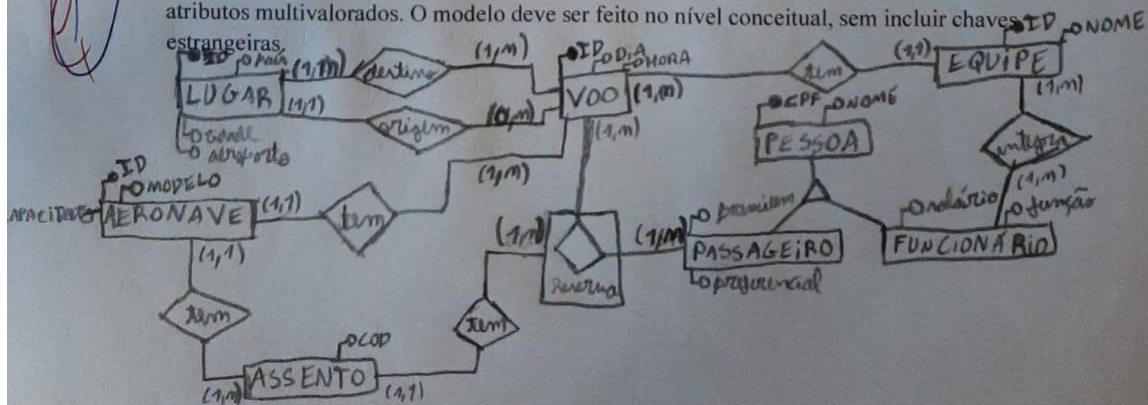
Médico(ID, nome) ✓

Paciente(CPF, nome, idade) ✓

Medicamento(ID, nome, farmaceutica, valor) ✓

Prescreve(#ID-med, #CPF-pac, #ID-remedio)

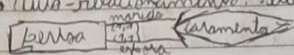
Questão 5. (4,0) Modele um banco de dados para um sistema de reservas de passagens aéreas, levando em consideração informações sobre voos, passageiros, assentos e reservas. Cada reserva está associada a um voo específico e a um passageiro. Através de um diagrama Entidade-Relacionamento, deve ser modelada esta base de dados. A base de dados não deve conter redundância de dados. O modelo ER deve ser representado com a notação vista em aula. O modelo deve apresentar, ao menos, entidades, relacionamentos, atributos, especializações, identificadores e restrições de cardinalidade. Não usar atributos multivalorados. O modelo deve ser feito no nível conceitual, sem incluir chaves estrangeiras.



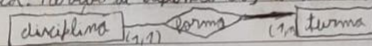
ER → entidade-relacionamento → modelo conceitual
 → abstrato, independente de SGBD

Elemento	Função	Gramática	Simbologia
Entidade	Agrupar, classificar	Substantivos	Entidade
Atributo	Características, propriedades	Adjetivos	→ atributo
Relacionamento	Associação	Verbo	→ relação em cardinalidade
Agregação	Relacionamentos de relacionamentos	—	→ agregação
Especialização	Divisão de propriedades	—	→ especialização

• Auto-relacionamento: relacionamento da entidade consigo mesma. É preciso definir o papel.



• Entidade fraca: relação de dependência, uma só existe quando relacionada à outra.



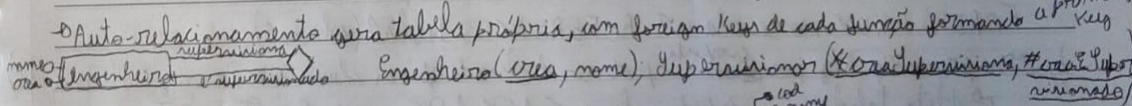
• Cardinalidade: (min, max) no relacionamento. Ex: gorgom (1,1) atende (1,1) marcar.
 → rel: gorgom atende de 1 a 1 marcar, e uma marca é atendida por 1 gorgom.

* Modelo Lógico → independente de SGBD, descreve as estruturas as tabelas em detalhes

Ordem: 1. traduz entidades e seus atributos / 2. traduz relacionamentos, na seguinte ordem: auto, entidade fraca, limitaria, término, agregação. / 3. traduz generalizações

→ Entidades viram tabelas, atributos viram colunas. Ex: Pessoa (id, nome, data nascimento)

Auto-relacionamento para tabela própria, com foreign keys de cada função formando a primary key.



→ Entidade fraca recebe a chave primária da entidade forte.

Disciplina (cod, nome) Turma (#coddisc, letra, horario, sala)

→ Relacionamentos limitarios 1:1 → 1 tabela; 1:n → adiciona coluna do 1 no n; m:n → 2 tabelas

Ex: Emp (1,1) Emp (1,1) → Empregado (cod, empregado, setor)

Ex: Gorgom (1,1) marca (1,1) cliente → Gorgom (cod, nome) Mera (num, qtd cliente, #cod-gorgom)

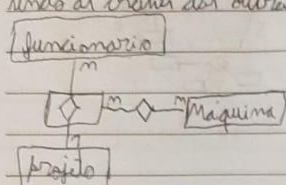
Ex: Prof (1,1) Escola (1,1) → Prof (cod, nome) Escola (cod, nome) Trabalha (cod prof, #cod esc)

→ Limitaria viria tabela da relação

Part (emp, razão social) Licença (cod, uf, nome) Produto (cod, valor) Participa (#cod lic, #cod prod, #emp)

Cardinalidade n viria chave primária, cardinalidade 1 é só estrangeira

→ Agregação cria tabelas normalmente e por fim adiciona tabela para agregação com chave primária sendo as chaves das outras tabelas.



funcionario(¹cpf, nome, ²#codproj)

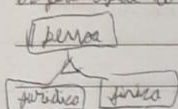
Projeto (cod, nome)

Máquina (cod, disc)

Utilizador(³#cpf, ³#codproj, ³#codmaq)

↳ com esta tabela fk codproj em funcionario ^{proprio} pode ser deletado

→ especialização pode tanto criar uma tabela p/ cada entidade (cada especialização recebe fk do principal como pk) ou uma tabela p/ todas, com uma coluna chamada tipo.



1) Pessoa (cod, nome)

~~funcionario~~ (#codkey, cpf, rg) Juiz (#codkey, cpf)

2) Pessoa (cod, nome, cpf, rg, cpf, tipo)

↳ dados podem ser nulos

* Modelo Físico → específico do SGBD (DDL → data definition language)

CREATE DATABASE; CREATE SCHEMA

CREATE TABLE nome (

nome-coluna1 char(1) NOT NULL,

nome-coluna2 int NOT NULL PRIMARY KEY,

FOREIGN KEY nome-coluna1 REFERENCES tabela1 (coluna1)

); ^{↳ não primary key por conta} pode ser declarada após as colunas com primary key (coluna, etc)

↳ delete tabela

↳ pode incluir CASCADE ou RESTRICT

DROP TABLE nome;

↳ apaga referências à esta tabela (foreign key) ^{↳ apaga se não houver dependência. É o padrão}

ALTER TABLE nome

drop nome-coluna1, -- delete a coluna com o respectivo nome

add coluna-nome integer NOT NULL,

add constraint pk em foreign key (coluna nome) references tabela70 (cod)

INSERT INTO tabela (cod, nome) values (15, 'Jefferson');

UPDATE tabela SET nome = 'João' WHERE cod = 15;

DELETE FROM cidade WHERE cod = 15